The following claims are presented for examination:

1. (currently amended) A method comprising:

receiving a first plurality of protocol data units at a first input of a protocol-data-unit excisor, wherein all of the protocol data units received at said first input are *en route* to a first congestible node;

receiving at [[a]] **said** protocol-data-unit excisor a metric of a queue in [[a]] **said** first congestible node; and

selectively dropping, at said protocol-data-unit excisor, one or more **of said first plurality of** protocol data units ~~en route to said first congestible node~~ based on said metric of said queue in said first congestible node.

2. (previously presented) The method of claim 1 wherein said protocol-data-unit excisor decides whether to drop a protocol data unit based on Random Early Detection.

3. (currently amended) The method of claim 1 further comprising:

receiving a second plurality of protocol data units at a second input of said protocol-data-unit excisor, wherein all of the protocol data units received at said second input are *en route* to a second congestible node;

receiving at said protocol-data-unit excisor a metric of a queue in [[a]] **said** second congestible node; and

selectively dropping, at said protocol-data-unit excisor, one or more **of said second plurality of** protocol data units ~~en route to said second congestible node~~ based on said metric of said queue in said second congestible node.

4. (currently amended) A protocol-data-unit excisor comprising:

a first input for receiving a first plurality of protocol data units, wherein all of the protocol data units received at said first input are *en route* to a first congestible node;

a **second input** ~~receiver~~ for receiving a metric of a queue in [[a]] **said** first congestible node; and

a processor for selectively dropping ~~, at said protocol-data-unit excisor,~~ one or more **of said first plurality of** protocol data units ~~en route to said first congestible node~~ based on said metric of said queue in said first congestible node.

5. (previously presented) The protocol-data-unit excisor of claim 4 wherein said protocol-data-unit excisor decides whether to drop a protocol data unit based on Random Early Detection.

6. (currently amended) The protocol-data-unit excisor of claim 4 further comprising:

a third input for receiving a second plurality of protocol data units, wherein all of the protocol data units received at said third input are _en route_ to a second congestible node;

a **fourth input** ~~receiver~~ for receiving a metric of a queue in [[a]] **said** second congestible node; [[and]]

[[a]] **wherein said** processor **is also** for selectively dropping ~~, at said protocol-data-unit excisor,~~ one or more **of said second plurality of** protocol data units ~~en route to said second congestible node~~ based on said metric of said queue in said second congestible node.

7. (currently amended) A method comprising:

receiving a first plurality of protocol data units at a first input of a protocol-data-unit excisor, wherein all of the protocol data units received at said first input are _en route_ to a first congestible node;

~~observing at a protocol-data-unit excisor the flow of protocol data units en route to a first congestible node;~~

estimating **in said protocol-data-unit excisor** a **first** metric of a **first** queue of protocol data units in said first congestible node based on said ~~flow~~ **first plurality** of protocol data units; and

selectively dropping, at said protocol-data-unit excisor, one or more **of said first plurality of** protocol data units _en route_ to said first congestible node based on said **first** metric ~~of said queue of protocol data units in said first congestible node~~ .

8. (previously presented) The method of claim 7 wherein said protocol-data-unit excisor decides whether to drop a protocol data unit based on Random Early Detection.

9. (currently amended) The method of claim 7 further comprising:

receiving a second plurality of protocol data units at a second input of said protocol-data-unit excisor, wherein all of the protocol data units received at said second input are _en route_ to a second congestible node;

~~observing at said protocol-data-unit excisor the flow of protocol data units~~ ~~*en route* to a second congestible node;~~

estimating **in said protocol-data-unit excisor** a **second** metric of a **second** queue of protocol data units in said second congestible node based on said ~~flow~~ **second plurality** of protocol data units; and

selectively dropping, at said protocol-data-unit excisor, a **one or more of said second plurality of** protocol data [[unit]] **units** *en route* to said second congestible node based on said **second** metric ~~of said queue of protocol data units in said second congestible node~~ .

10. (currently amended) A protocol-data-unit excisor comprising:

a first input for receiving a first plurality of protocol data units, wherein all of the protocol data units received at said first input are *en route* to a first congestible node; and

~~a transmitter arranged to observe the flow of protocol data units *en route* to a first congestible node; and~~

a processor for estimating a **first** metric of a **first** queue of protocol data units in said first congestible node based on said [[flow]] **first plurality** of protocol data units, and for selectively dropping one or more **of said first plurality of** protocol data units *en route* to said first congestible node based on said ~~metrics of said queue~~ **first metric**.

11. (previously presented) The protocol-data-unit excisor of claim 10 wherein said processor for selectively dropping one or more protocol data units decides whether to drop a protocol data unit based on Random Early Detection.

12. (currently amended) The protocol-data-unit excisor of claim 10 further comprising:

a second input for receiving a second plurality of protocol data units, wherein all of the protocol data units received at said second input are *en route* to a second congestible node; and

~~a transmitter arranged to observe the flow of protocol data units *en route* to a second congestible node; and~~

a processor for estimating a **second** metric of a **second** queue of protocol data units in said second congestible node based on said [[flow]] **second plurality** of protocol data units, and for selectively dropping one or more **of said second plurality of** protocol data

units *en route* to said second congestible node based on said **second** metric ~~of said queue of protocol data units in said second congestible node~~ .